

Package ‘isaeditor’

September 29, 2021

Title Tools to Manipulate ISA-Tab Files

Version 0.1.1

Description ISA-Tab (Investigation/Study/Assay (ISA) tab-delimited (TAB) format) is a general purpose framework for storing complex metadata in omics applications. It is notoriously hard to manipulate due to the fact that it is a graph rather than a tab-delimited data frame. The 'isaeditor' package is meant to facilitate reading, writing, displaying, manipulating, modifying and populating ISA-Tab files in R.

License GPL (>= 3)

URL <https://github.com/bihealth/isaeditor/>

Encoding UTF-8

RoxygenNote 7.1.1

Imports colorDF, crayon, tibble, dplyr, glue, tidyr, readr, magrittr, methods, rlang

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author January Weiner [aut, cre] (<<https://orcid.org/0000-0003-1438-7819>>)

Maintainer January Weiner <january.weiner@gmail.com>

Repository CRAN

Date/Publication 2021-09-29 08:00:02 UTC

R topics documented:

isatab-class	2
isa_ID_find	5
isa_nodes	6
isa_node_add	7
isa_properties	8
isa_rows_add	9
node_list	10

n_row	11
read_isa	11

Index	13
--------------	-----------

isatab-class	<i>Class for assay and study objects</i>
--------------	--

Description

Class for isatab assay and study objects

Usage

```
## S3 method for class 'isatab'
dim(x)

## S3 method for class 'isatab'
print(x, ...)

## S3 method for class 'isatab'
as.data.frame(x, ...)

## S3 method for class 'isatab'
as_tibble(x, ...)

## S3 method for class 'isatab'
n_row(x)

## S3 method for class 'isatab'
summary(object, ...)

## S3 replacement method for class 'isatab'
x[node, property = NULL, new = FALSE, n = NA, after_id = NULL] <- value

## S3 method for class 'isatab'
x[node, property = NULL, n = NA]

## S3 replacement method for class 'isatab'
x[[col_id]] <- value

## S3 method for class 'isatab'
x[[col_id]]
```

Arguments

x	object of class isatab
...	any further arguments are ignored

object	object of class isatab
node	node column (e.g. 'Sample Name')
property	property column (e.g. 'Performer')
new	force creating a new node even if there is already a node with such an identifier
n	instance of the node identifier (if there are multiple identical node identifiers in the isatab, for example multiple 'Extract Name' nodes).
after_id	ID of an existing column. If a column (node or property) needs to be created, after_id can be used to specify after which node / column the new column will be inserted.
value	vector or data frame with values which will be inserted into the isatab at the specified column.
col_id	Column ID (e.g. 'ID34')

Details

Objects of this class are generated usually by reading a file with `read_isa()`.

Internally, it is a list containing as elements a data frame (tibble) describing the structure of the isa tab (`isa_stru`) and a data frame (tibble) containing the actual data.

Terminology:

ISA-tab *nodes* (such as 'Source Name', 'Sample Name', 'Protocol REF', 'Extract Name' or 'Library Name') can have *properties*. Both are represented as *columns*. In the ISA-tab specification, node designators such as 'Sample Name' are called *identifiers*, although they need not be unique. IDs are internal identifiers of the package `isaeditor`; they are unique to a column. Some functions in `isaeditor` can access ISA-tab columns using node / property combination; some other require the internal ID.

Accessing columns (nodes and properties) of an isa tab:

Note: IDs are a thing internal to this R package. They are not imported from or exported to actual ISA-tab files. However, given that the node 'identifiers' (e.g. 'Sample Name') can be ambiguous, IDs are necessary to unambiguously identify a node.

There are two ways of accessing a column: by using the `[]` function to select a node identifier (e.g. 'Protocol REF') and, optionally, a property identifier (e.g. 'Performer'), or by using the `[[` function to select column IDs. The former has the disadvantage that multiple identical node / property identifier combinations may exist, and it may be necessary to specify which node is meant:

```
isa_a <- read_isa('a_isatab.txt')
isa_a[ 'Sample Name' ]
isa_a[ 'Protocol REF', 'Performer' ]
## 3rd instance of the combination Protocol REF / Performer
isa_a[ 'Protocol REF', 'Performer', n=3 ]
isa_a[ 'Protocol REF', 'Performer', n=3 ] <- 'Rosalind Franklin'
```

Assigning a NULL value to a selected node is equivalent to removing this node and all its properties.

Assigning a NULL value to a property is equivalent with removing this property.

Using column IDs with the `[[` function is not ambiguous, but column IDs are a trick used by the package `isaeditor` and are not exported or read from an actual ISA-tab. To view the column IDs, simply print the `isatab` object to the screen or inspect the `isa_stru` element of the object:

```
isa_s <- read_isa('s_isatab.txt')
isa_s
isa_s$isa_stru
isa_s[['ID21']]
isa_s[['ID21']] <- 'Rosalind Franklin'
```

Both `[` and `[[` return a vector if a single column is specified and a data frame if more than one column is selected.

Creating and removing nodes and properties:

Nodes and properties can either be created with `isa_node_add()` and `isa_property_add()` or with assigning a value to a new node with `[<-`:

```
isa_a['Test Node'] <- c(1, 2, 3)
isa_a['Test Node', 'Test Property'] <- 5:7
```

In the above code, first the node `Test Node` was created and filled with values `1:3`, and then the property `Test Property` was created and filled with `5:7`. This can be shortened by assigning a data frame in one step:

```
isa_a['Test Node', 'Test Property'] <- data.frame(1:3, 5:7)
```

A column ID can be specified to insert the node at a position relative to another node, or the property at a position relative to another property:

```
isa_a[ 'Test Node', after_id='ID1' ] <- 1:3
```

Removing nodes and properties works by assigning `NULL` to either a node (in which case all node properties will be removed as well) or a property.

```
# remove only one property
isa_a['Test Node', 'Test Property'] <- NULL
# remove node and its properties
isa_a['Test Node'] <- NULL
```

Value

An object of `isatab-class` is a list containing three elements:

- `isa_stru`, a data frame holding the meta-data
- `contents`, a data frame holding the data
- `type`, the type of the `isatab` component (study, investigation, assay).

See Also

`read_isa()` `isa_ID_find()`

isa_ID_find	<i>Find IDs of nodes or properties</i>
-------------	--

Description

Find IDs of nodes or properties fulfilling specified criteria

Usage

```
isa_ID_find(x, node_pattern = NULL, value_pattern = NULL, prop_pattern = NULL)
```

Arguments

x	object of class isatab
node_pattern	return only nodes which match the given pattern
value_pattern	return only nodes which match one of the values
prop_pattern	return only nodes which match one of the properties

Details

Note: IDs are a thing internal to this R package. They are not imported from or exported to actual ISA-tab files. However, given that the node 'identifiers' (e.g. 'Sample Name') can be ambiguous, IDs are necessary to unambiguously identify a node.

Value

Character vector of IDs

See Also

[isatab](#)

Examples

```
file <- system.file('extdata',  
  'a_isatab.txt',  
  package='isaeditor')  
isa_a <- read_isa(file)  
isa_ID_find(isa_a, node_pattern='.* Name')  
isa_a[['ID34']]
```

isa_nodes	<i>Show nodes in an isatab</i>
-----------	--------------------------------

Description

Show nodes in an isatab

Usage

```
isa_nodes(x)
```

Arguments

x object of class isatab

Details

Note: IDs are a thing internal to this R package. They are not imported from or exported to actual ISA-tab files. However, given that the node 'identifiers' (e.g. 'Sample Name') can be ambiguous, IDs are necessary to unambiguously identify a node.

Value

Returns a data frame (tibble) containing columns with node ID, node identifier (name), number of properties associated with that node and a summary of the values for that node.

See Also

[isatab](#)

[isatab, isa_properties\(\)](#)

Examples

```
file <- system.file('extdata', 's_isatab.txt', package='isaeditor')
isa_s <- read_isa(file)
isa_nodes(isa_s)
```

isa_node_add	<i>Add or remove nodes and properties</i>
--------------	---

Description

Add or remove nodes and properties

Usage

```
isa_node_add(x, node, columns = NULL, after_node = NULL)
```

```
isa_node_rm(x, node_id)
```

```
isa_property_add(x, property, values = NA, node_id = NULL, after_id = NULL)
```

```
isa_property_rm(x, prop_ids = NULL)
```

Arguments

x	isatab object
node	new node identifier (e.g. 'Sample Name')
columns	(optional) character vector with columns to add
after_node	ID of the node after which the current node should be inserted
node_id	ID of the node in which to add the property (default: last node in the isatab).
property	Character vector with identifiers (such as 'Comment[Important]') of the properties to be inserted
values	vector (if only one property is added) or data frame (if multiple properties are added) of values used to initialize the node / parameter. If multiple properties are added with one call (the length of the property vector is greater than one), and values is a data frame, then it has to have sufficient number of columns corresponding to the property vector.
after_id	ID of the property after which the parameter should be inserted (default: last property)
prop_ids	IDs of the properties to be removed

Details

These functions manipulate the structure of an isatab. `isa_node_add` and `isa_node_rm` add or remove whole nodes.

To add or remove properties (individual columns which are not nodes) belonging to a given node, use `isa_property_add` and `isa_property_rm`.

Adding and removing nodes is easier using brackets / subscripts. Read the documentation for [isatab](#) for details.

Note: IDs are a thing internal to this R package. They are not imported from or exported to actual ISA-tab files. However, given that the node 'identifiers' (e.g. 'Sample Name') can be ambiguous, IDs are necessary to unambiguously identify a node.

Value

These functions return an object of isatab-class.

See Also

[isatab](#)

Examples

```
file <- system.file('extdata', 's_isatab.txt', package='isaeditor')
isa_s <- read_isa(file)
isa_s <- isa_node_add(isa_s, 'Library Name', columns='Comment[Raw File]')
isa_nodes(isa_s)
isa_s <- isa_property_add(isa_s, 'Characteristics[Age]', values=c(75, 38, 43), node_id='ID1')
```

isa_properties

Show properties associated with a node ID

Description

Show properties associated with a node ID

Usage

```
isa_properties(x, node_id)
```

Arguments

x	object of class isatab
node_id	ID of a node

Details

Note: IDs are a thing internal to this R package. They are not imported from or exported to actual ISA-tab files. However, given that the node 'identifiers' (e.g. 'Sample Name') can be ambiguous, IDs are necessary to unambiguously identify a node.

Value

Returns a named character vector. Names are the IDs of properties associated with a given node, and values are the property names.

See Also

[isatab](#), [isa_nodes\(\)](#)

Examples

```
file <- system.file('extdata', 's_isatab.txt', package='isaeditor')
isa_s <- read_isa(file)
isa_properties(isa_s, 'ID1')
```

isa_rows_add	<i>Add sample rows to an isatab</i>
--------------	-------------------------------------

Description

Add sample rows to an isatab

Usage

```
isa_rows_add(x, n, total = FALSE, replicate = TRUE)
```

Arguments

x	an isatab object
n	number of rows to add
total	if TRUE, the resulting isatab object will have n rows.
replicate	If true (default), the values in the last row of the isatabs will be replicated. Otherwise, empty rows will be added.

Details

Expand the isatab by adding rows. If total parameter is TRUE, the number of rows to be added will be such that the final number of rows is n. However, if n is smaller than current number of rows, no rows will be removed.

Value

An object of class isatab with expanded rows

See Also

[isatab](#)

Examples

```
file <- system.file('extdata', 's_isatab.txt', package='isaeditor')
isa_s <- read_isa(file)
isa_new <- isa_rows_add(isa_s, 10, total=TRUE)
n_row(isa_new)
```

`node_list`*Lists all the nodes in an isatab object*

Description

Lists all the nodes in an isatab object

Usage

```
node_list(x)
```

```
node_show(x, node_id)
```

```
node_select(x, node_id, inverse = FALSE)
```

```
prop_select(x, prop_id, inverse = FALSE)
```

Arguments

<code>x</code>	object of class <code>isatab</code>
<code>node_id</code>	ID of a node to show
<code>inverse</code>	if TRUE, inverses the selection
<code>prop_id</code>	property IDs to be selected

Details

`node_list` returns a data frame with one row per node, showing the number of properties associated with a given node and a summary of values for that node.

`node_show` returns a data frame for a given `node_id` listing all properties associated with that node and a summary of values for each of the properties.

`node_select` returns a new object of class `isatab` containing only the selected nodes.

`prop_select` returns a new object of class `isatab` containing only the selected property IDs (which may not be node IDs!).

Value

Functions `node_list` and `node_show` return a `data.frame` like object. Functions `node_select` and `prop_select` return an object of class `isatab`.

Examples

```
file <- system.file('extdata', 's_isatab.txt', package='isaeditor')
isa_s <- read_isa(file)
node_list(isa_s)
```

n_row	<i>Generic replacement for nrow()</i>
-------	---------------------------------------

Description

Generic replacement for nrow()

Usage

```
n_row(x)
```

Arguments

x an array-like object

Value

an integer of length 1 or NULL.

read_isa	<i>Read or write an isatab file</i>
----------	-------------------------------------

Description

Read or write an isatab file

Usage

```
read_isa(file, type = "auto")
```

```
write_isa(x, file)
```

Arguments

file file name to read / write
type Either 'auto', or 'investigation', 'study', 'assay' (can be abbreviated)
x isatab object

Value

read_isa() returns either an object of class isatab (for study / assay files) or an object of class isa_i (for investigation files).

See Also

[isatab](#)

Examples

```
file <- system.file('extdata', 'i_Investigation.txt', package='isaeditor')
isa_i <- read_isa(file)
print(isa_i)
```

```
file <- system.file('extdata', 's_isatab.txt', package='isaeditor')
isa_s <- read_isa(file)
print(isa_s)
```

Index

`[.isatab (isatab-class), 2`
`[<-.isatab (isatab-class), 2`
`[[.isatab (isatab-class), 2`
`[[<-.isatab (isatab-class), 2`

`as.data.frame.isatab (isatab-class), 2`
`as_tibble.isatab (isatab-class), 2`

`dim.isatab (isatab-class), 2`

`isa_ID_find, 5`
`isa_ID_find(), 4`
`isa_node_add, 7`
`isa_node_add(), 4`
`isa_node_rm (isa_node_add), 7`
`isa_nodes, 6`
`isa_nodes(), 8`
`isa_properties, 8`
`isa_properties(), 6`
`isa_property_add (isa_node_add), 7`
`isa_property_add(), 4`
`isa_property_rm (isa_node_add), 7`
`isa_rows_add, 9`
`isatab, 5–9, 11`
`isatab-class, 2`

`n_row, 11`
`n_row.isatab (isatab-class), 2`
`node_list, 10`
`node_select (node_list), 10`
`node_show (node_list), 10`

`print.isatab (isatab-class), 2`
`prop_select (node_list), 10`

`read_isa, 11`
`read_isa(), 3, 4`

`summary.isatab (isatab-class), 2`

`write_isa (read_isa), 11`